



Relatório Técnico

**Núcleo de
Computação Eletrônica**

Bon Marché

**Um Grupo de
Agentes Móveis para o
Comércio na Internet**

Rosane Maria Martins

Luci Pirmez

Luiz Fernando Rust da Costa Carmo

NCE - 02/99

Universidade Federal do Rio de Janeiro

Bon Marché
Um Grupo de Agentes Móveis para o Comércio na Internet

Rosane Maria Martins*, Luci Pirmez e Luiz Fernando Rust da Costa Carmo
rosane@eng.uerj.br, {luci, rust}@nce.ufrj.br
NCE/UFRJ - Núcleo de Computação Eletrônica
Universidade Federal do Rio de Janeiro
Tel.: 021 598-3159 - Caixa Postal: 2324 - Rio de Janeiro RJ Brasil

Resumo

A abertura da Internet para uso comercial com sua interface gráfica de fácil uso gerou um enorme crescimento na quantidade de usuários e variedade de serviços disponíveis nestes últimos anos. Atualmente, grande parte das instituições públicas e privadas fornecem algum tipo de informação ou serviço através da Internet.

Tendo em vista este cenário, o presente trabalho propõe o desenvolvimento de um software auxiliar multiagente, móvel e inteligente para o mercado eletrônico visando tornar as compras na Internet mais fáceis para o usuário. O sistema proposto é baseado em uma sociedade de agentes que tentam encontrar os produtos pretendidos pelo usuário em paralelo nas várias lojas que estão a eles associadas, apresentando os resultados de uma maneira uniforme e organizada. Para tanto, serão utilizados uma técnica de Inteligência Artificial que se baseia em um mecanismo de inferências a partir de regras e o ambiente para o desenvolvimento de sistemas de agentes móveis, Aglets Software Development Kit, desenvolvido pela IBM Tóquio.

Este artigo discute a arquitetura deste projeto e ressalta os resultados obtidos em uma primeira fase onde foi construído um protótipo de configuração mais simples que implementa a mobilidade. Numa segunda fase, será implementado o módulo de inteligência o qual tornará o grupo de agentes Bon Marché mais autônomo e capaz de realizar consultas e recomendações mais refinadas.

Abstract

The beginning of Internet as commercial usage with its easy graphic interface originated an enormous increase of users' quantity as well as a great variety of available services during the last years. Nowadays the majority of private and public institutions are prepared to provide any kind of information or service through the Internet.

Considering this fact, the present work proposes the development of a mobile and intelligent multiagent auxiliary software to the electronic market in order to make easier the Internet's purchases. The proposed system is based on a group of agents that try to find simultaneously the users' interest products in the several virtual places known by them, presenting the results in an homogeneous way. The techniques applied to the development of the system will be the IBM's mobile agent framework and the rule-driven inferences mechanism.

This paper discusses the architecture of the project and emphasizes, in a first phase, the results obtained with a prototype built with a configuration more simplified that implements the mobility. In a second phase, it will be built the intelligent module to transform the Bon Marché group of agents into a more autonomous and able group to achieve more appurated queries and recommendations.

* Aluna do curso de Mestrado em Informática do Instituto de Matemática da Universidade Federal do Rio de Janeiro e funcionária da Universidade do Estado do Rio de Janeiro, lotada no Departamento de Engenharia de Sistemas e Computação

1 - Introdução

O processo de compras, atualmente, na Internet é muito semelhante àquele utilizado nas compras convencionais. Em primeiro lugar, para comprar um produto é necessário encontrar uma loja on-line que forneça esse produto. Uma vez encontrado o fornecedor, é preciso procurar o produto dentro da loja. Este processo terá que ser repetido para várias lojas até que o produto seja encontrado com as melhores características possíveis, tendo como limite o tempo que o comprador pretende despende no processo de compra.

O primeiro problema com que se depara um potencial comprador na Internet é, muitas vezes, a inexistência de *software* que o auxilie neste processo, de modo que o comprador tentará encontrar diretamente o produto com a ajuda de mecanismos de busca. Por uma série de razões, é muito pouco provável que por este processo obtenha alguma informação relevante, apesar de se obter muitas respostas.

Utilizando o processo de compra análogo ao convencional, o comprador consegue obter informação relevante, mas o processo é demorado e repetitivo. Além disso, a comparação entre produtos não é fácil, porque os produtos são apresentados pelas várias lojas em formatos diferentes, e essa apresentação não é simultânea. Como resultado, é freqüente que o comprador acabe por se contentar com algo não desejado por não querer perder tempo demais.

Muito do tempo útil de trabalho humano, que acaba sendo perdido com tarefas simples como esta, poderia ser realizado por assistentes inteligentes junto à Internet, assistentes que teriam a finalidade de auxiliar o usuário na rede. Cada assistente inteligente constitui-se em um agente modular e autônomo, e encontra seu embasamento teórico na Inteligência Artificial e na Inteligência Artificial Distribuída respectivamente.

Os sistemas de Inteligência Artificial tradicionais concentravam sua atenção em um único agente, o qual seria provido de alguma espécie de inteligência, e sozinho seria especialista na realização de uma tarefa específica.

Atualmente o enfoque tende a ser outro, fala-se em Inteligência Artificial Distribuída. Nesta sub-área de Inteligência Artificial o enfoque recai sobre o aspecto de integração de agentes providos de alguma inteligência. São especialistas na atividade que desempenham, não trabalham mais isolados, mas de forma cooperativa tentam resolver um problema da melhor forma possível, caracterizando uma área da Inteligência Artificial Distribuída, a área dos Sistemas Multiagentes.

O termo agente é muito utilizado por diferentes áreas, como sistemas distribuídos, engenharia de software entre outras, possuindo, portanto, diferentes definições. Entretanto, os agentes apresentam algumas características próprias que os diferenciam dos demais softwares.

Os agentes são entidades autônomas, e encontram-se, geralmente, imersos em uma sociedade. Conceitualmente são entidades capazes de reagir e / ou agir a favor de algo ou alguém, influenciando seu ambiente. Este ambiente necessariamente inclui outros agentes. Interagir com este ambiente, ou mais especificamente com outros agentes, é o elemento chave de um Sistema Multiagente.

O presente trabalho propõe o desenvolvimento de um sistema multiagente, móvel e inteligente para o mercado eletrônico visando tornar as compras através da Internet mais fáceis para o usuário. Isso é obtido melhorando a fase do processo de compra em que o usuário tenta obter informação sobre os produtos ou a gama de produtos já selecionados, de forma a decidir qual a melhor alternativa para a satisfação da sua necessidade.

Os agentes de compras tentam encontrar os produtos pretendidos em paralelo nas várias lojas na Internet que estão a eles associadas. As informações obtidas são, então, apresentadas de forma uniformizada e organizada. Com os dados assim apresentados pelo sistema, é muito mais fácil para o usuário escolher o produto com as características mais satisfatórias.

Ao utilizar o sistema, o comprador:

- deixará de ter que se preocupar com as diferentes interfaces das lojas, já que a interface será uma só durante todo o processo de procura dos produtos;
- passará a ver os produtos e as suas características num formato que facilite a sua comparação;
- poupará tempo e dinheiro, porque o agente de compras, além de ser mais rápido em fazer as procuras (já que as poderá efetuar em paralelo em várias lojas conhecidas), pode efetuá-las sem que o usuário esteja on-line;
- deixará de ter o tempo como um fator limitativo às suas compras, pois o agente de compras não se importará de fazer uma pesquisa exaustiva.

Para a busca dos produtos desejados, será utilizada pelos agentes, uma técnica de inteligência artificial que se baseia em um mecanismo de inferências a partir de regras. Os vários agentes, que serão associados a cada loja disponível no sistema, comunicar-se-ão entre si e migrarão entre servidores utilizando o framework oferecido pelo *Aglets Software Development Kit(ASDK)*, um ambiente para o desenvolvimento de sistemas de agentes móveis desenvolvido pela IBM Tokyo Research Laboratory, baseado em Java 1.1.

Este trabalho está organizado da seguinte forma: a seção 2 apresenta uma descrição dos conceitos e técnicas utilizados no trabalho; na seção 3 é apresentada em detalhes a arquitetura do sistema proposto e, finalmente, a conclusão será dada na seção 4.

2 - Conceitos e Técnicas Utilizados

Nesta seção serão apresentados os conceitos iniciais relacionados à Inteligência Artificial Distribuída e aos agentes e, por fim, será feita uma descrição sucinta dos ambientes para o desenvolvimento de agentes existentes, focalizando o *Aglets Software Development Kit(ASDK)* da IBM Tokyo, por ser o ambiente utilizado no desenvolvimento desse trabalho.

2.1 - Inteligência Artificial Distribuída

A Inteligência Artificial Distribuída (IAD) constitui-se em uma das áreas de pesquisa da Inteligência Artificial tradicional. A IAD procura estudar o conhecimento e as técnicas de raciocínio que podem ser úteis para que o grupo de agentes computacionais integrem uma sociedade de agentes.

A IAD desenvolve técnicas de Inteligência Artificial para tentar solucionar problemas de maneira distribuída. Essas técnicas passam a ser úteis na solução de problemas amplos onde o domínio da questão é inerentemente distribuído no espaço, como, por exemplo, pode-se citar os problemas de controle de tráfego aéreo, distribuição de energia elétrica, etc.

Em IAD, entidades individuais que contribuem para solução de um problema são chamadas de *agentes*. Agentes quando agrupados podem formar uma comunidade do tipo cooperativa com o objetivo de atingir uma meta comum (solucionar um problema global).

Pode-se dividir o mundo da Inteligência Artificial Distribuída em dois enfoques principais: a Resolução Distribuída de Problemas (RDP) e Sistemas Multiagentes (SMA) [2],[1]. Vários autores citam esta divisão no mundo da IAD, mas é importante observar que a linha que separa as duas áreas de pesquisa ainda é muito tênue, ou seja, não é muito clara entre estas duas áreas.

Na *Resolução Distribuída de Problemas*, o enfoque principal é o *problema*. Seus objetivos principais são fazer uso da capacidade de processamento oferecida pela tecnologia de redes para atacar problemas naturalmente distribuídos, como a decomposição e a alocação de tarefas em uma rede de computadores. Os agentes envolvidos em RDP são programados para cooperar, dividir tarefas, comunicar-se de maneira confiável, etc.

Nos *Sistemas Multiagentes*, a pesquisa preocupa-se com a coordenação do comportamento inteligente entre uma coleção de agentes autônomos inteligentes, nos seguintes aspectos: como eles podem coordenar seus conhecimentos, objetivos, habilidades e planejar juntamente como realizar ações ou solucionar problemas. Os agentes em um Sistema Multiagente podem compartilhar conhecimentos sobre problemas e soluções, abordando, também, os processos de coordenação entre os agentes. Nesta área os agentes já existem, e o problema é distribuído para que estes o resolvam.

2.2 - Agentes

A Inteligência Artificial Distribuída (IAD) aborda a atividade de um agente autônomo em um ambiente multiagente. Ainda não existe uma definição uniforme para o termo agente dentro da IAD. A seguir, serão fornecidas três possíveis definições:

1. *"Chama-se de agente uma entidade abstrata que é capaz de agir sobre ela mesma e sobre seu próprio ambiente, que dispõe de uma representação parcial deste ambiente, e que, em um universo multiagentes, pode comunicar-se com outros agentes, e cujo comportamento é consequência de suas observações, de seu comportamento e das interações com outros agentes".*[4]
2. *"Um agente é uma entidade a qual pode-se associar uma entidade única, e que é capaz de realizar cálculos formais. Um agente pode ser considerado como um meio que produz um certo número de ações a partir dos conhecimentos e mecanismos internos que lhe são próprios."* [6]
3. *"Agente é a palavra utilizada para designar uma entidade "inteligente" e autônoma. A palavra autônoma refere-se ao fato de que cada agente possui sua própria existência, a qual não é dependente da existência de outros agentes."* [5]

Deve-se notar no caso da última definição que o atributo de inteligência não é necessariamente uma propriedade útil para caracterizar e distinguir agentes de outros tipos de *software* que existem no mercado, como, por exemplo, os Tutores Inteligentes. Para alguns, o termo agente apenas significa entidades "autônomas" que atuam em nome de alguém ou próprio, para atingir uma determinada meta. Para outros autores, um agente é necessariamente "inteligente" e "autônomo", sendo que a autonomia é um pouco mais detalhada pelos aspectos de requerer ação periódica, execução espontânea, e iniciativa, sendo que os agentes devem ter habilidade de realizar ações independentes e antecipar-se a ações, as quais irão beneficiar o usuário[10].

A IAD baseia-se no comportamento social, com ênfase nas ações e nas interações dos agentes, os quais podem ser desde extremamente simples até extremamente complexos. Um comportamento social inteligente pode surgir no caso de membros "inteligentes" da sociedade (conhecidos como Agentes Cognitivos) ou no caso de membros não "inteligentes" da sociedade (conhecidos como Agentes Reativos) [5].

Como as qualidades consideradas desejáveis em um agente de *software*, pode-se citar:

- **Autonomia:** um agente é habilitado a tomar iniciativa e exercer um grau não trivial de controle sobre suas próprias ações.

- **Orientação a Meta:** um agente aceita pedidos de alto nível, e é responsável por decidir um grau não trivial de controle sobre suas próprias ações.
- **Flexibilidade:** as ações de um agente não seguem um roteiro, estes são habilitados a realizar escolhas dinâmicas de quais ações realizar, e em qual sequência, em resposta a um estado do ambiente externo.
- **Inicialização Própria:** diferente dos programas padrões que são diretamente disparados pelos usuários, um agente pode perceber modificações em seu ambiente e decidir quando atuar.
- **Comunicação:** um agente é habilitado a ocupar-se com uma comunicação complexa com outros agentes, incluindo pessoas, com o intuito de obter informações ou conseguir ajuda para cumprir seus objetivos.
- **Inferência Social:** um agente deve ser capaz de inferir sobre as atividades de outros agentes.
- **Inteligência:** habilidade que os agentes possuem para raciocinar e aprender a partir das interações com outros agentes, com seus proprietários e com o ambiente. Um agente pode ser dotado de diferentes níveis de inteligência. Níveis mais baixos incluem a capacidade de lidar com sentenças de preferências do usuário, as quais podem ser apresentadas sob a forma de regras, com um mecanismo de inferência ou algum mecanismo de raciocínio atuando nessas preferências. Níveis mais altos de inteligência incluem um modelo do usuário e alguma outra forma de entendimento e raciocínio sobre o que um usuário quer fazer, e o planejamento dos meios de obter essa meta. No topo da escala de inteligência estão os sistemas que aprendem e adaptam-se ao seu ambiente, tanto em termos dos objetivos dos usuários quanto dos recursos disponíveis.
- **Mobilidade:** a característica da mobilidade corresponde à capacidade do agente se mover no ambiente. Quando se trata de agentes de software, um agente móvel é aquele que é capaz de se transportar de uma máquina para outra durante sua execução.
- **Continuidade Temporal:** agentes são processos que estão em execução continuamente, não como um instante computacional que mapeia uma única entrada para uma única saída, e termina.

É importante enfatizar que os agentes para executarem suas tarefas podem ou não usar as capacidades supracitadas; e é em função dessas capacidades que eles podem receber outras denominações, tais como: agentes fixos, agentes móveis, agentes inteligentes, agentes móveis e inteligentes, entre outras.

Os agentes podem ser classificados também de acordo com a tarefa que executam. Atualmente, algumas das possíveis aplicações (ainda que se trate, em alguns casos, apenas de protótipos) em que estão sendo usados os agentes são:

- filtragem de informação, como, por exemplo, de mensagens de correio eletrônico e de "news";
- recuperação de informação (permitindo, entre muitas outras coisas, encontrar páginas pessoais na WWW);
- gerência de redes (monitoramento, filtragem e notificação);
- o comércio eletrônico (por exemplo, permitindo a utilização de vários métodos de pagamento através de uma só interface ou ajudando o usuário a encontrar o produto que pretende);
- a educação e o treinamento;

Existem ainda outras possibilidades para efetuar a classificação, nomeadamente, de acordo com o ambiente em que se inserem: agentes de software (em mundos virtuais), robots (se o ambiente for físico) entre outros.

2.2.1 - Agentes Móveis

As definições de agentes móveis seguem um padrão, visto que ressaltam a possibilidade de migração de código e estado, bem como a autonomia de execução.

Em [3], "agentes móveis são programas que têm a capacidade de migrar entre máquinas de uma rede durante a sua execução, carregando consigo seu estado de execução:

agente = comportamento, estado & localização"

Em [7], "um agente móvel é definido como aquele que pode se mover em uma rede heterogênea, de um nó para outro, sobre seu próprio controle, interagindo com recursos existentes, ou com outros agentes, tipicamente retornando ao local de origem quando sua tarefa estiver concluída."

Eles oferecem um novo e importante método de realizar transações e recuperar informações em redes. Entretanto, os agentes móveis introduzem novos cuidados com a segurança.

O recente avanço na área de agentes móveis tem sido impulsionado pelas deficiências do modelo cliente / servidor em aplicações distribuídas. O modelo cliente / servidor impõe regras fixas aos programas e um conjunto fixo de operações especializadas no servidor. Além disso, no caso do servidor não possuir a instrução específica, o cliente poderá realizar várias consultas intermediárias, gastando desnecessariamente a banda de rede.

Técnicas modernas associadas ao modelo cliente / servidor permitem um programa migrar para o servidor. Essas técnicas, contudo, ainda mantêm a divisão entre o cliente e o servidor. Os programas estão limitados a trocar resultados parciais com quem os chamou, mantendo sempre uma conexão entre as máquinas.

Agentes móveis são um paradigma mais próximo para a computação distribuída. Eles fazem uso mais eficiente dos recursos da rede, são semelhantes ao modelo "peer-to-peer" e suportam desconexões da rede, sem interromper sua execução. Além disso, eles permitem ao usuário trabalhar em um alto nível de abstração, no qual os canais de comunicação ficam transparentes para o mesmo. Os agentes móveis evitam o problema de operações especializadas no servidor, porque o código do agente migra para o local onde recursos estão localizados, evitando desperdícios de tráfego na rede de dados intermediários.

2.3 - Ambientes para o desenvolvimento de Agentes

Há um grande número de pacotes para a construção de agentes no mercado que permitem ao usuário construir e gerenciar seus próprios agentes. Felizmente, o advento da linguagem Java tem gerado uma enxurrada de sistemas experimentais. Numerosos ambientes estão ainda em desenvolvimento e alguns deles disponíveis na web para avaliação.

Serão descritos nesta seção alguns sistemas ou ambientes desenvolvidos para dar suporte à implementação de sociedades de agentes. Entretanto, será dado um enfoque maior ao Aglets Software Development Kit (ASDK), ambiente usado neste projeto.

O pacote Tabriz AgentWare da General Magic foi o pioneiro, e provavelmente, o mais conhecido ambiente que executa e gerencia aplicações baseadas em agente nos servidores. Existe, também, a ferramenta Tabriz Agent Tools para criação de agentes destinados aos "sites" da web.

Já a arquitetura JAT (Java Agent Template) foi desenvolvida por Rob Frost da Universidade de Stanford e fornece um modelo funcional completo (escrito inteiramente utilizando a linguagem Java), para a construção de agentes de software que comunicam-se dois a dois, com uma comunidade de outros agentes distribuídos através da Internet. Embora o código que define cada um seja portátil, os agentes projetados no JAT, não são móveis (migratórios), tendo uma existência fixa em um determinado *host* [9].

A IBM desenvolveu dois pacotes: ABE (Agente Building Environment) e Aglets Software Development Kit (ASDK). A ferramenta ABE facilita a construção de uma aplicação baseada em agentes inteligentes, além de fornecer condições para incrementar aplicações existentes com agentes. O ASDK, desenvolvido nos laboratórios de pesquisa da IBM situados no Japão, auxilia na

criação de agentes móveis[6]. O pacote completo oferece um ambiente gráfico para construção de aplicações em Java que exigem a implementação de tais agentes, além de um servidor de agente (Tahiti) e a especificação do Agent Transfer Protocol (ATP).

O ASDK estende o modelo de applets de Java (*aglet* = *agent* + *applet*) e possui como principais componentes:

- uma plataforma de execução (servidor de aglets);
- Agent Transfer Protocol (ATP) - protocolo para transmissão de aglets;
- ATP daemons - processos que recebem requisições ATP em cada máquina;
- Aglet-API (A-API) - conjunto de classes e interfaces base com primitivas para a interação com o lugar e outros agentes;
- Fuji - browser WWW usado para interface com o usuário.

O modelo ASDK define um conjunto de abstrações e o comportamento necessário à implementação da tecnologia de agente móvel na Internet. Seus conceitos chave são: *aglet*, *proxy*, *context*, *identifier*:

- **Aglet:** Uma aglet é um objeto móvel em Java habilitado a visitar servidores na rede. É autônomo, porque roda com sua própria linha de execução após chegar ao servidor destino, e reativo, porque pode responder a mensagens.
- **Proxy:** Um proxy é o representante de uma aglet. Serve como um escudo que protege a aglet de acessos externos aos seus métodos públicos. O proxy também fornece transparência de localização para a aglet, isto é, ele pode esconder a real localização da aglet.
- **Context:** Um contexto de uma aglet é o local de trabalho da mesma. É um objeto estacionário que fornece um meio de manutenção e manipulação da aglet em um ambiente de execução uniforme, onde o sistema servidor está protegido de aglets "desonestas".
- **Identifier:** Um identificador é o limite para cada aglet. Este identificador é único e imutável durante todo o ciclo de vida de uma aglet.

3 - Descrição do Sistema

Agentes para o comércio eletrônico que serão considerados neste contexto são agentes que ajudam, de alguma forma, o usuário a fazer compras através da Internet. Este tipo de agente, designado por agente de compras, pode desempenhar diversas tarefas, como por exemplo: ajudar o usuário a decidir que produto comprar; fazer sugestões, com base no seu conhecimento sobre seu

proprietário; descobrir novidades, descontos e preços especiais; encontrar as lojas que vendem o produto ou serviço desejado, entre outras coisas.

Das possíveis funções descritas acima, o Agente Bon Marché pretende ajudar a encontrar as lojas que vendem o produto desejado e a listar os preços dos produtos encontrados. Sendo assim, o objetivo do trabalho proposto é tornar o processo de compras através da Internet mais fácil para o usuário, já que torna desnecessário a procura de vários fornecedores para o produto pretendido repetidas vezes. Além disso, uma vez que o processo de procura do produto passa a ser substancialmente mais rápido e pode ainda ser realizado "off-line", o tempo disponível para fazer compras deixa de ser um fator limitativo e o usuário passa a encontrar mais vezes o produto que quer com o melhor preço.

Para se alcançar este objetivo, o funcionamento do Agente Bon Marché é o seguinte:

- o usuário escolhe o tipo de produto específica, se o desejar, as características desejadas para esse produto (características essas que serão as restrições às condições de procura);
- o agente de compras procura, entre os produtos desse tipo, aqueles com as características desejadas (a procura pode ser efetuada sem o usuário estar "on-line", sendo os resultados visualizados mais tarde);
- como resultado de pesquisa, o agente Bon Marché envia um e-mail ou mostra uma página HTML ao usuário com uma lista de produtos, seus respectivos preços e onde encontrá-los.

3.1 - Arquitetura Proposta:

Para que o agente Bon Marché tenha a funcionalidade supracitada e possa, no futuro, ser estendido para muitos produtos e muitas lojas a seguinte arquitetura é proposta:

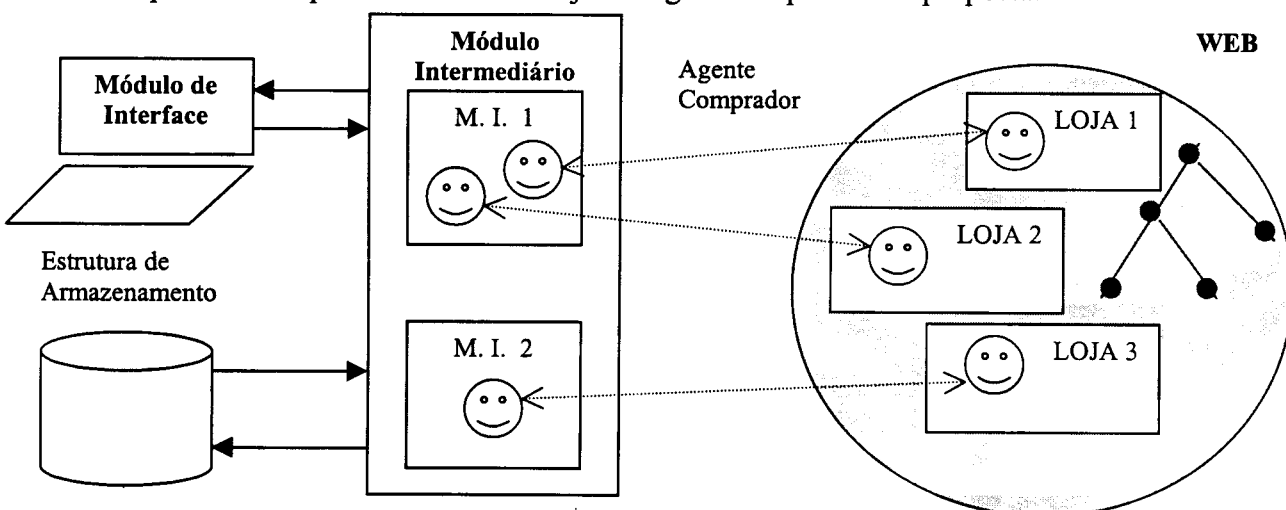


Figura 1 - Arquitetura Geral do Agente de Compras Bon Marché

Conforme ilustrado na figura 1, o sistema apresentado é composto pelos seguintes componentes: Módulo de Interface, Módulo Intermediário, Agente Comprador e Estrutura de Armazenamento. A seguir, um maior detalhamento sobre os componentes do sistema:

- **Módulo de Interface:** é o componente pelo qual o usuário entra em contato com o sistema e realiza o seu pedido. Módulo também responsável por apresentar ao usuário o resultado obtido pelo conjunto de agentes.

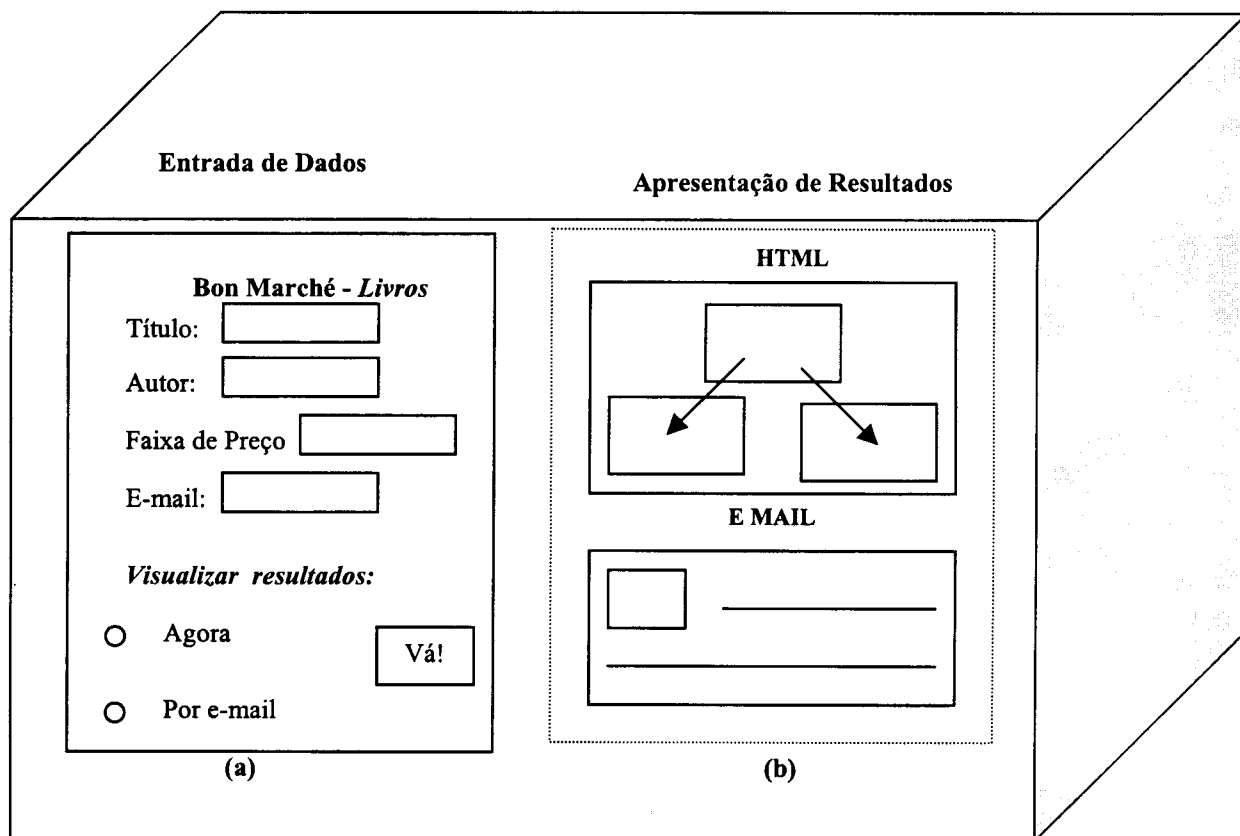


Figura 2 - Estrutura Interna do Módulo de Interface

Conforme ilustrado na figura 2(a), "título", "autor" e "faixa de preço" são informações que o usuário deve fornecer ao Módulo de Interface para que ele possa solicitar ao Módulo Intermediário a criação e o disparo dos agentes compradores de acordo com as restrições impostas pelo usuário.

Assim que o módulo gerenciador de resultados (componente do módulo intermediário) agrega todas as respostas obtidas, ele envia ao módulo de interface para que as mesmas sejam entregues ao usuário: ou por meio de páginas HTML ou por e-mail, conforme a figura 2(b).

- **Módulo Intermediário:** Módulo responsável pela criação e lançamento dos agentes compradores para dar início à pesquisa solicitada pelo comprador. Agrega, também, os resultados obtidos pelos agentes. Existe um módulo intermediário para cada tipo de produto disponível no sistema, como, por exemplo, um módulo intermediário para livros e outro para CDs.

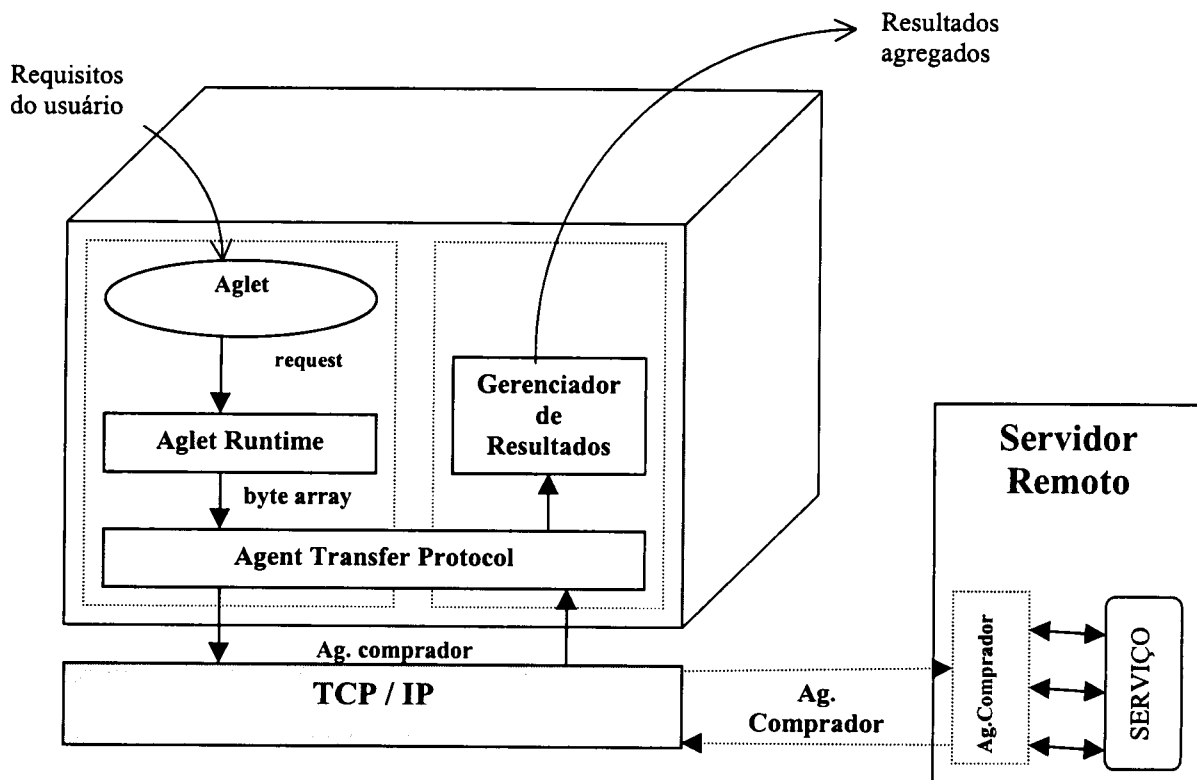


Figura 3 - Estrutura Interna do Módulo Intermediário

Este módulo controla o funcionamento do sistema como um todo, gerenciando todas as informações obtidas dos vários agentes. É responsável efetivamente pela criação, envio e recebimento dos agentes.

Como pode ser visto na figura 3, ao receber os requisitos do usuário do módulo de interface, o Módulo de Intermediário, através do seu componente “Gerenciador de Agentes” cria os agentes de acordo com esses requisitos, e envia-os para os endereços disponíveis na Estrutura de Armazenamento.

Quando o Módulo Intermediário recebe uma solicitação para disparar um agente, este último é criado na camada “aglet” e viaja pela camada de “runtime”, a qual converte-o em um array de bytes e este array, por sua vez, é passado à camada do ATP – Agent Transfer Protocol para que seja enviado ao seu destino. Este protocolo constrói, então, um “bit stream” que contém tanto

informações gerais, tais como: o nome do sistema e a identificação do mesmo, como o “byte array” proveniente da camada de “runtime”.

Cada agente comprador ao retornar do servidor com os dados de sua pesquisa envia todo o seu conteúdo ao componente Gerenciador de Resultados (Módulo Intermediário), para que este possa agregar todas as respostas obtidas e enviá-las ao módulo de interface.

- ***Agentes Compradores:*** Realizam o contato com a loja através de suas páginas na Web, efetuam o pedido e interpretam as páginas com as respostas geradas, convertendo-as em um formato entendido pelo módulo intermediário. Os agentes antes de seguirem para o seu destino são codificados em “bit stream”: os primeiros segmentos são informações gerais, tais como, a identificação do agente e o último segmento é o “byte-array” que é o agente propriamente dito: código e estado. Têm como meta a verificação das páginas contidas no seu endereço destino, selecionando aquelas consideradas relevantes e recomendadas de acordo com as regras pré-estabelecidas. Elas irão formar a base de regras a ser utilizada pelo agente para tomar decisões apropriadas no processo de avaliação dos itens encontrados.

Exemplo: Considerando que as informações fornecidas pelo usuário tenham sido:

Título: “Mobile Agents” ***Autor:*** “Cockayne” ***Faixa de Preço:*** “50”

As regras, neste caso, poderiam ser:

Regra 1:

if título = "Mobile Agents" and autor = "Cockayne"
and preço ≤ 50
then statusLivro= "relevante"

Regra2:

if título = "Mobile Agents" and autor= "Cockayne"
and preco > 50
then statusLivro= "relevante"

Regra3:

if título = "Mobile Agents"
then statusLivro= "recomendado"

Regra4:

if título = "Agents"
and preco ≤ 50
then statusLivro= "recomendado"

Regra5:**if** título = "Mobile"**and** preco < =50**then** statusLivro= "recomendado"

A lógica de raciocínio a ser utilizada será o encadeamento para frente: "Forward Chaining". De acordo com o valor da variável statusLivro é que o agente decidirá se irá enviar as informações encontradas ao componente gerenciador de resultados ou não.

Com esta arquitetura, a extensão do agente de compras para tratar de novos produtos e novas lojas é simples, apesar de ser necessário construir um módulo intermediário para cada novo produto.

- **Estrutura de Armazenamento:** estrutura *hash* que contém os endereços das diversas lojas associadas ao sistema. Para cada tipo de produto pesquisado pelo sistema existe uma estrutura desta. Para cada loja dessa base de dados, o módulo intermediário lançará um agente comprador. Esses agentes terão conhecimento do formato em que a loja fornece as respostas. Assim, terão a capacidade de extrair dessas respostas a informação de que necessitam.

Estrutura para o Módulo Intermediário de Livros:

Exemplo.:

Livraria ABC	http://www.livrariaABC.com.br
BuyBook	http://www.BuyBook.com

4 - Conclusões

O trabalho aqui apresentado abordou uma das muitas facetas do tratamento da informação no ambiente da Internet. Este artigo enfocou principalmente o desenvolvimento de um sistema multiagente móvel e inteligente para a realização de consultas refinadas visando reduzir o tempo gasto no processo de compra via Internet. O presente trabalho se diferencia das demais propostas existentes na literatura por aliar a mobilidade à Inteligência Artificial Distribuída.

Atualmente, o projeto apresenta uma configuração mais simples, permitindo a existência de apenas um tipo de produto a ser pesquisado, um fornecedor e um agente comprador. O produto escolhido para esta fase foi "livro". O agente comprador é criado na máquina do usuário e migra para o servidor aglet onde se encontra o serviço oferecido pelo fornecedor.

A implementação do módulo inteligente e a possibilidade da existência de mais de um tipo de produto a ser pesquisado tarefas a serem implementadas em uma segunda fase da construção deste projeto.

Concluindo, pode-se afirmar que, apesar do projeto se encontrar numa fase ainda experimental, onde novas idéias têm sido constantemente avaliadas tanto no plano conceitual como no algorítmico, os resultados obtidos nesta primeira fase podem ser considerados satisfatórios , devido ao fato da filtragem das informações solicitadas pelo usuário ser realizada no servidor onde os recursos estão localizados, promovendo, assim, uma redução significativa do tráfego na rede.

REFERÊNCIAS BIBLIOGRÁFICAS:

1. BITTENCOURT, Guilherme. **Inteligência Artificial – Ferramentas e Teorias**. Campinas: Instituto de Computação, UNICAMP, 1996. Trabalho apresentado na Escola de Computação, 1996, Campinas, SP.
2. BOND, A.H.; GASSER , Les.(Eds) **Readings in Distributed Artificial Intelligence**. San Mateo, California: Morgan Kaufmann, 1988.
3. ENDLER, Markus. **Novos Paradigmas de Interação usando Agentes Móveis**, São Paulo, 1998. XVI Simpósio Brasileiro de Redes de Computadores.
4. FERBER, Jacques; GASSER, Les (Eds) **Intelligence Artificielle Distribuée- EC2**, Avignon 1991. Tutorial Notes of 11th Conference on Expert Systems and their applications (Avignon 1991)
5. FROZZA, Rejane. **Resolução de Problemas utilizando Sistemas Multiagentes**. Porto Alegre: CPGCC da UFRGS, 1995.
6. GASSER, Les. **Boundaries, identity and aggregation**. Plurality issues in multi-agent systems. In *Decentralized A.I.*, 3, WERNER, Eric, DEMAZEN, Yves (Eds), pp. 199-212, Amsterdam, Elsevier Science Publishers, 1992.
7. GRAY, R.; KOTZ, D.; NOG, S.; Rus, D.; CYBENKO, G. **Mobile Agents for Mobile Computing** – Technical Report PCS-TR96-285 – Department of Computer Science – Dartmouth College – 1996.
8. IBM TOKYO RESEARCH LABORATORY, **Aglets Specification Version 1.0 (Draft)** Abril 1998. (disponível em <http://www.trl.ibm.com/documents.html>)
9. Java Agent Template. Disponível em: <http://cdr.stanford.edu/ABR/JavaAgent.html>
10. PETRIE, Charles J.; **Agent-Based Engineering, the Web, and Intelligence**; <http://cdr.stanford.edu/NextLink/Expert.html>